## Array属性:Length（LongLength）

### 1.格式：int[ ] a = new int [ 3 ] ;

### Console.WriteLine(a.Length);

### 2.用户可以获取数组的长度，对于一维数组来说，通过“数组名.Length”就可以得到数组的长度

### 3.Length获取一个32位的整数，该值表示所有维度的数组中的元素总数

### 4.LongLength获取一个64位整数，该值表示所有维度的数组中的元素总数

## Array属性：Rank

### 1.格式：public int Rank [ get； ]

### 2.Rank获取数组的秩（维度）

## Array属性：IsReadOnly

### 1.获取一个值，该值指示数组是否只读

## Array属性：IsFixedSize

### 1.获取一个值，该值指示数组是否带有固定大小

## Array常见的方法

### Add方法：添加

### Insert方法方法：中间插入

|  |
| --- |
| https://img2018.cnblogs.com/blog/1653349/201904/1653349-20190426092158895-1269961191.png |

### Remove方法:删除

### RemoveAt:删除指定索引的元素

### RemoveRange:删除指定数量的元素

### Clear方法:清空

|  |
| --- |
| https://img2018.cnblogs.com/blog/1653349/201904/1653349-20190426091943160-149555.png |

### Sort方法:排序,默认是升序

### Reverse方法:反转数组

|  |
| --- |
| https://img2018.cnblogs.com/blog/1653349/201904/1653349-20190426092436359-1850727074.png |

### Copy方法复制数组

|  |
| --- |
| https://img2018.cnblogs.com/blog/1653349/201904/1653349-20190426100818570-1094052132.png |
| https://img2018.cnblogs.com/blog/1653349/201904/1653349-20190426100908779-1863560267.png |

### GetLength方法:获取元素个数

### GetType方法:获当前实例的类型

### GetValue方法:获取指定索引的值

### SetValue方法:给指定位置的元素设置值

### IndexOf方法,方法数组在第一次出现的索引.

### ToString方法:获取当前对象的字符串描述

### GetUpperBound用于获取Array的指定维度的上限

### GetLowerBound用于获取Array的指定维度的下限

|  |
| --- |
| string[,] abcd = new string[2, 4] //很明显这是一个2\*4的数组  //就是第一个数组中包含4个元素，第二个数组也包含4个元素，第三个数组也包含4个元素  //Rank就是维数 这里代表的就是[2,4]中的2 下面的循环是依次遍历每一个数组中包含的数组  for (int i = 0; i < abcd.Rank; i++)  {  //GetUpperBound方法 获取 Array 的指定维度的上限。下面这段代码就是遍历每个数组每 一个元素  for (int j = 0; j <= abcd.GetUpperBound(abcd.Rank - 1); j++)  }  [复制代码](javascript:void(0);)  [复制代码](javascript:void(0);)  Console.WriteLine("狼人杀游戏主要身份：");//提示信息  //定义数组，存储狼人杀游戏主要角色  string[] roles = { "狼人", "预言家", "村民", "女巫", "丘比特", "猎人", "守卫\n" };  foreach(string role in roles)//遍历数组  {  Console.Write(role + " ");//输出遍历到的元素  }  Console.WriteLine("数组的个数：");  Console.WriteLine(roles.Length);//输出为7  Console.WriteLine(roles.GetLowerBound(0));//输出为0  Console.WriteLine(roles.GetUpperBound(0));//输出为6  Console.ReadLine(); |